APPLICATION FOR UNITED STATES LETTERS PATENT

FOR

TWO-DIMENSIONAL PYRAMID FILTER ARCHITECTURE

Inventor(s): Tinku Acharya

Prepared by: Howard Skaist,
Senior IP Attorney

intel®
Intel Corporation
2111 N. E. 25th Avenue; JF3-147
Hillsboro, OR 97124
Phone: (503) 264-0967
Facsimile: (503) 264-1729

TWO-DIMENSIONAL PYRAMID FILTER ARCHITECTURE

This application is a continuation of Application No. 09/846,609, filed April 30, 2001.

RELATED APPLICATIONS

This patent application is related to U.S. Patent Application Serial No. 09/754,684, titled "Multiplierless Pyramid Filter," filed January 3, 2001, by Tinku Acharya; U.S. Patent Application Serial No. 09/817,711, titled "Two-Dimensional Pyramid Filter Architecture," (attorney docket 042390.P11275), filed March 26, 2001, by Tinku Acharya; U.S. Patent Application Serial No. 09/820,108, titled "Pyramid Filter," (attorney docket 042390.P11211), filed March 28, 2001, by Tinku Acharya; U.S. Patent Application Serial No. 09/823,212, titled "Two-Dimensional Pyramid Filter Architecture," (attorney docket 042390.P11276), filed March 30, 2001, by Tinku Acharya, and U.S. Patent Application No. 09/823,390, titled "Two-Dimensional Pyramid Filter Architecture," (attorney docket 042390.P11277), filed March 30, 2001, by Tinku Acharya, all assigned to the assignee of the presently claimed subject matter and herein incorporated by reference.

BACKGROUND

This disclosure is related to pyramid filters.

In image processing, it is often desirable to decompose an image, such as a scanned color image, into two or more separate image representations. For example, a color or gray-scale document image can be decomposed into background and foreground images for efficient image processing operations, such as enhancement, compression, etc., as are at times applied in a typical photocopying machine or scanner device. In this context, this operation is often referred to as a descreening operation. This descreening is also sometimes applied to remove halftone patterns that may exist in an original scanned image. For example, these halftone patterns may cause objectionable artifacts for human eyes if not properly removed. The traditional approach for this decomposition or descreening is to filter the color image in order to blur it. These blurred results are then used to assist in determining how much to blur and sharpen the image in order to produce the decomposition. Typically this blurring can be achieved using a "symmetric pyramid" filter. Symmetric pyramid finite impulse response (FIR) filters are well-known.

One disadvantage of this image processing technique, however, is that the complexity increases many fold when a number of pyramid filters of different sizes is applied in parallel in order to generate multiple blurred images, to apply the technique as just described. A brute force approach for this multiple pyramid filtering approach is to use multiple FIR filters in parallel, as illustrated in FIG. 1. Such an approach demonstrates that the design and implementation of fast "symmetric pyramid filtering" architectures to generate different blurred images in parallel from a single source image may be desirable.

The numbers provided in parenthesis for each FIR block in FIG. 1 represents the pyramid filter of corresponding length. For example, (1, 2, 1) are the filter coefficients for a symmetric pyramid finite impulse response (FIR) filter of order or length 3. Likewise, (1, 2, 3, 2, 1) are the coefficients for an FIR pyramid filter of order 5, (1, 2, 3, 4, 3, 2, 1) are the coefficients for an FIR pyramid filter of order 7, (1, 2, 3, 4, 5, 4, 3, 2, 1) are the coefficients for an FIR pyramid filter of order 9, (1, 2, 3, 4, 5, 6, 5, 4, 3, 2, 1) are the coefficients for an FIR pyramid filter of order 11, and so forth.

Unfortunately, the approach demonstrated in FIG. 1 has disadvantages. For example, inefficiency may result from redundant computations. Likewise, FIR implementations frequently employ multiplier circuits. While implementations exist to reduce or avoid the use of multipliers, such as with shifting and summing circuitry, that may then result in increased clocking and, hence, may reduce circuit throughput. A need, therefore, exists for improving pyramid filtering implementations or architectures.

BRIEF DESCRIPTION OF THE DRAWINGS

Subject matter is particularly pointed out and distinctly claimed in the concluding portion of the specification. The claimed subject matter, however, both as to organization and method of operation, together with objects, features, and appendages thereof, may best be understood by reference of the following detailed description when read with the accompanying drawings in which:

FIG. 1 is a block diagram illustrating a brute force approach to implementing a finite impulse response (FIR) multiple pyramid filtering architecture;

FIG. 2 is one embodiment of a one-dimensional multiplierless pyramid filter;

FIG. 3 is one embodiment of a two-dimensional pyramid filter architecture;

FIG. 4 is a table/matrix showing an example of a matrix that may result from implementing a two-dimensional pyramid filter architecture, such as one that may be implemented by the embodiment of FIG. 3;

FIG. 5 is a table/matrix showing an example of a two-dimensional signal that may be operated upon by a two-dimensional pyramid filter architecture;

FIG. 6 is a table/matrix showing an example of applying a one-dimensional pyramid filter kernel both row-wise and column-wise;

FIG. 7 is the table/matrix of FIG. 6 for k=11;

FIG. 8 is a table/matrix showing the result of applying a one-dimensional pyramid filter to the rows of a two-dimensional input signal sample matrix; and

FIG. 9 is a table/matrix showing the result of applying a one-dimensional pyramid filter to the columns of a two-dimensional input signal sample matrix.

DETAILED DESCRIPTION

In the following detailed description, numerous specific details are set forth in order to provide a thorough understanding of the claimed subject matter. However, it will be understood by those skilled in the art that the claimed subject matter may be practiced without these specific details. In other instances, well-known methods, procedures, components and circuits have not been described in detail in order so as not to obscure the claimed subject matter.

As previously described, pyramid filtering, in particular, symmetric pyramid filtering, may be employed in connection with color images or color image processing in order to decompose or descreen the image, such as into a background and foreground image, for example. Although the claimed subject matter is not limited in scope in this respect, in such a context, pyramid filtering architectures that reduce computational complexity or processing and/or hardware cost are particularly desirable. Likewise, implementations that are multiplerless, that is, do not specifically employ multiplication in the implementation, are also desirable usually because such implementations or embodiments are cheaper to implement than those that employ or include multiplier circuits.

Although the claimed subject matter is not limited in scope in this respect, FIG. 2 illustrates an embodiment 200 of a one-dimensional pyramid filter, such as described in more detail in aforementioned U.S. Patent Application Serial No. 09/754,684, titled "Multiplierless Pyramid Filter," by T. Acharya (attorney docket no. 042390.P10722), filed on January 3, 2001. Embodiment 200 comprises a unified multiplierless cascaded symmetric pyramid filtering architecture to generate a multiple number of filtered output signal streams for a series or sequence of pyramid filters having different orders, the generation of the output signal streams occurring in parallel. In this particular embodiment, although, again, the claimed subject matter is not limited in scope in this respect, a filtered output signal stream is produced on every clock cycle for each pyramid filter of a different order being implemented. Therefore, in addition to being computationally efficient, this particular embodiment produces good results in terms of throughput. However, as previously indicated, this particular embodiment implements a one-dimensional pyramid filter.

FIG. 2 is understood in the context of specific notation. For example, an input source signal, X, may be designated as follows:

$$X = (x_0, x_1, \ldots, x_{i-2}, x_{i-1}, x_i, x_{i+1}, x_{i+2}, \ldots.)$$

In digital or discrete signal processing, filtering may be expressed as a convolution, $\otimes$, of the input signal, X, and a filter, F, in this context a digital filter of finite length, referred to here as a finite impulse response (FIR) filter. Therefore, the filtered output signal stream is indicated as follows:

$$Y = X \otimes F$$

As previously described, the particular embodiment in FIG. 2 employs pyramid filters. These filters are typically implemented using digital filters of lengths or orders that are odd, such as 3, 5, 7, 9, etc. Odd numbers or orders, in this context, may be expressed in the form 2N-1, where N is a positive integer greater than two, for example. Some examples of such digital filters are as follows:

$$F_3 = (1, 2, 1)$$

$$F_5 = (1, 2, 3, 2, 1)$$

$$F_7 = (1, 2, 3, 4, 3, 2, 1)$$

$$F_9 = (1, 2, 3, 4, 5, 4, 3, 2, 1)$$

$$F_{11} = (1, 2, 3, 4, 5, 6, 5, 4, 3, 2, 1)$$

...

$F_M = (1, 2, 3, \ldots, N, \ldots, 3, 2, 1)$ (where, in this context, $M = 2N-1$)

For the foregoing filters, the filtered output signals or output signal streams may be represented as follows:

$B^3 = X \otimes F_3 = (b_0^3, b_1^3, \ldots, b_{i-1}^3, b_i^3, b_{i+1}^3, \ldots.)$ result of input signal X filtered by F_3

$B^5 = X \otimes F_5 = (b_0^5, b_1^5, \ldots, b_{i-1}^5, b_i^5, b_{i+1}^5, \ldots.)$ result of input signal X filtered by F_5

$B^7 = X \otimes F_7 = (b_0^7, b_1^7, \ldots, b_{i-1}^7, b_i^7, b_{i+1}^7, \ldots.)$ result of input signal X filtered by F_7

$B^9 = X \otimes F_9 = (b_0^9, b_1^9, \ldots, b_{i-1}^9, b_i^9, b_{i+1}^9, \ldots.)$ result of input signal X filtered by F_9

$B^{11} = X \otimes F_{11} = (b_0^{11}, b_1^{11}, \ldots, b_{i-1}^{11}, b_i^{11}, b_{i+1}^{11}, \ldots.)$ result of input signal X filtered by F_{11}

.....

$B^M = X \otimes F_M = (b_0^M, b_1^M, \ldots, b_{i-1}^M, b_i^M, b_{i+1}^M, \ldots.)$ result of input signal X filtered by F_M

An alternate way to empirically represent these filtered output signal samples is as follows:

$b_i^3 = x_{i-1} + 2x_i + x_{i+1}$

$b_i^5 = x_{i-2} + 2x_{i-1} + 3x_i + 2x_{i+1} + x_{i+2}$

$b_i^7 = x_{i-3} + 2x_{i-2} + 3x_{i-1} + 4x_i + 3x_{i+1} + 2x_{i+2} + x_{i+3}$

$b_i^9 = x_{i-4} + 2x_{i-3} + 3x_{i-2} + 4x_{i-1} + 5x_i + 4x_{i+1} + 3x_{i+2} + 2x_{i+3} + x_{i+4}$

$b_i^{11} = x_{i-5} + 2x_{i-4} + 3x_{i-3} + 4x_{i-2} + 5x_{i-1} + 6x_i + 5x_{i+1} + 4x_{i+2} + 3x_{i+3} +2x_{i+4} +x_{i+5}$

Likewise, by introducing what is referred to, in this context, as state variables, the above expressions may be re-expressed as follows:

$b_i^3 = x_i + s_i^3$, where $s_i^3 = x_{i-1} + x_i + x_{i+1}$

$b_i^5 = b_i^3 + s_i^5$, where $s_i^5 = x_{i-2} + x_{i-1} + x_i + x_{i+1} + x_{i+2}$

$b_i^7 = b_i^5 + s_i^7$, where $s_i^7 = x_{i-3} + x_{i-2} + x_{i-1} + x_i + x_{i+1} + x_{i+2} + x_{i+3}$

$b_i^9 = b_i^7 + s_i^9$, where $s_i^9 = x_{i-4} + x_{i-3} + x_{i-2} + x_{i-1} + x_i + x_{i+1} + x_{i+2} + x_{i+3} + x_{i+4}$

$b_i^{11} = b_i^9 + s_i^{11}$, where $s_i^{11} = x_{i-5} + x_{i-4} + x_{i-3} + x_{i-2} + x_{i-1} + x_i + x_{i+1} + x_{i+2} + x_{i+3} + x_{i+4} +$
x_{i+5}

Hence, the desired pyramid filter may be expressed as follows:

$B^3 = X + S_3$, where $S_3 = (s_0{}^3, s_1{}^3, s_2{}^3, \ldots, s_{i-1}{}^3, s_i{}^3, s_{i+1}{}^3, \ldots)$

$B^5 = B^3 + S_5$, where $S_3 = (s_0{}^5, s_1{}^5, s_2{}^5, \ldots, s_{i-1}{}^5, s_i{}^5, s_{i+1}{}^5, \ldots)$

$B^7 = B^5 + S_7$, where $S_7 = (s_0{}^7, s_1{}^7, s_2{}^7, \ldots, s_{i-1}{}^7, s_i{}^7, s_{i+1}{}^7, \ldots)$

$B^9 = B^7 + S_9$, where $S_9 = (s_0{}^9, s_1{}^9, s_2{}^9, \ldots, s_{i-1}{}^9, s_i{}^9, s_{i+1}{}^9, \ldots)$

$B^{11} = B^9 + S_{11}$, where $S_{11} = (s_0{}^{11}, s_1{}^{11}, s_2{}^{11}, \ldots, s_{i-1}{}^{11}, s_i{}^{11}, s_{i+1}{}^{11}, \ldots)$

A study of FIG. 2 illustrates that the computed output signal streams, B_3, B_5, B_7, B_9, B_{11}, etc. of the pyramid filters shown in FIG. 2 are produced by the embodiment illustrated.

The previous discussion of pyramid filters occurs in the context of one-dimensional filtering; however, due at least in part to the symmetric nature of such filters, it is possible to implement pyramid two-dimensional filtering instead of computing in a row-wise and column-wise one-dimensional fashion that employs extra computational steps. If we represent the one-dimensional k -tap pyramid filter as $F_k = \begin{bmatrix} 1 & 2 & 3 & \cdots & \frac{k+1}{2} & \cdots & 3 & 2 & 1 \end{bmatrix}$, the corresponding two dimensional pyramid filter F_{kxk} may be derived as shown in FIG. 6. In FIG. 7, we have shown the two-dimensional pyramid filter kernel for k=11. Assuming a two-dimensional input signal, e.g., signal samples, having the form shown in FIG. 5, FIG. 4 is a table illustrating a matrix that may result, here a two-dimensional filtered signal sample output matrix, P^{kxk} , in which the two dimensional input signal sample matrix is filtered using two-dimensional pyramid filter kernel F_{kxk}.

The matrix shown in FIG. 8 may result from applying a one-dimensional k-tap pyramid filter in every row of the two-dimensional input signal sample matrix and the matrix shown in FIG. 9 may result from applying a one-dimensional k-tap pyramid filter in every column of the two-dimensional input signal sample matrix. The matrix in FIG. 4 may result from applying the two-dimensional (k x k) tap filter to the two dimensional input signal sample matrix or, alternatively, it may result from applying the one-dimensional k-tap pyramid filter row-wise and then followed by column-wise. Applying this approach to generate filtered signal sample outputs P^{1x3}, P^{3x1}, and P^{3x3}, produces the following relationships:

$$P_{i,j}^{1x3} = s_{i,j-1} + 2s_{i,j} + s_{i,j+1}$$

$$P_{i,j}^{3x1} = s_{i-1,j} + 2s_{i,j} + s_{i+1,j}$$

$$P_{i,j}^{3x3} = s_{i-1,j-1} + 2s_{i-1,j} + s_{i-1,j+1} + 2s_{i,j-1} + 4s_{i,j} + 2s_{i,j+1} + s_{i+1,j-1} + 2s_{i+1,j} + s_{i+1,j+1}$$

Generating filtered signal sample outputs P^{5x1}, P^{1x5}, and P^{5x5}, produces the following relationships:

$$P_{i,j}^{5x1} = \quad s_{i-2,j} + 2s_{i-1,j} + 3s_{i,j} + 2s_{i+1,j} + s_{i+2,j}$$

$$P_{i,j}^{1x5} = \quad s_{i,j-2} + 2s_{i,j-1} + 3s_{i,j} + 2s_{i,j+1} + s_{i,j+2}$$

$$\begin{aligned} P_{i,j}^{5x5} &= (s_{i-2,j-2} + 2s_{i-2,j-1} + 3s_{i-2,j} + 2s_{i-2,j+1} + s_{i-2,j+2}) + (2s_{i-1,j-2} + 4s_{i-1,j-1} + 6s_{i-1,j} + 4s_{i-1,j+1} + 2s_{i-1,j+2}) \\ &+ (3s_{i,j-2} + 6s_{i,j-1} + 9s_{i,j} + 6s_{i,j+1} + 3s_{i,j+2}) + (2s_{i+1,j-2} + 4s_{i+1,j-1} + 6s_{i+1,j} + 4s_{i+1,j+1} + 2s_{i+1,j+2}) \\ &+ (s_{i+2,j-2} + 2s_{i+2,j-1} + 3s_{i+2,j} + 2s_{i+2,j+1} + s_{i+2,j+2}) \end{aligned}$$

Likewise, generating filtered signal samples outputs P^{7x1}, P^{1x7}, and P^{7x7}, produces the following relationships:

$$P_{i,j}^{7x1} = s_{i-3,j} + 2s_{i-2,j} + 3s_{i-1,j} + 4s_{i,j} + 3s_{i+1,j} + 2s_{i+2,j} + s_{i+3,j}$$

$$P_{i,j}^{1x7} = s_{i,j-3} + 2s_{i,j-2} + 3s_{i,j-1} + 4s_{i,j} + 3s_{i,j+1} + 2s_{i,j+2} + s_{i,j+3}$$

$$\begin{aligned}P_{i,j}^{7x7} = &(s_{i-3,j-3} + 2s_{i-3,j-2} + 3s_{i-3,j-1} + 4s_{i-3,j} + 3s_{i-3,j+1} + 2s_{i-3,j+2} + s_{i-3,j+3}) + \\ &(2s_{i-2,j-3} + 4s_{i-2,j-2} + 6s_{i-2,j-1} + 8s_{i-2,j} + 6s_{i-2,j+1} + 4s_{i-2,j+2} + 2s_{i-2,j+3}) + \\ &(3s_{i-1,j-3} + 6s_{i-1,j-2} + 9s_{i-1,j-1} + 12s_{i-1,j} + 9s_{i-1,j+1} + 6s_{i-1,j+2} + 3s_{i-1,j+3}) + \\ &(4s_{i,j-3} + 8s_{i,j-2} + 12s_{i,j-1} + 16s_{i,j} + 12s_{i,j+1} + 8s_{i,j+2} + 4s_{i,j+3}) + \\ &(3s_{i+1,j-3} + 6s_{i+1,j-2} + 9s_{i+1,j-1} + 12s_{i+1,j} + 9s_{i+1,j+1} + 6s_{i+1,j+2} + 3s_{i+1,j+3}) + \\ &(2s_{i+2,j-3} + 4s_{i+2,j-2} + 6s_{i+2,j-1} + 8s_{i+2,j} + 6s_{i+2,j+1} + 4s_{i+2,j+2} + 2s_{i+2,j+3}) + \\ &(s_{i+3,j-3} + 2s_{i+3,j-2} + 3s_{i+3,j-1} + 4s_{i+3,j} + 3s_{i+3,j+1} + 2s_{i+3,j+2} + s_{i+3,j+3})\end{aligned}$$

Likewise, generating filtered signal samples outputs P^{9x1}, P^{1x9}, and P^{9x9}, produces the following relationships:

$$P_{i,j}^{9x1} = s_{i-4,j} + 2s_{i-3,j} + 3s_{i-2,j} + 4s_{i-1,j} + 5s_{i,j} + 4s_{i+1,j} + 3s_{i+2,j} + 2s_{i+3,j} + s_{i+4,j}$$

$$P_{i,j}^{1x9} = s_{i,j-4} + 2s_{i,j-3} + 3s_{i,j-2} + 4s_{i,j-1} + 5s_{i,j} + 4s_{i,j+1} + 3s_{i,j+2} + 2s_{i,j+3} + s_{i,j+4}$$

$$
\begin{aligned}
P_{i,j}^{9x9} = {} & (s_{i-4,j-4} + 2s_{i-4,j-3} + 3s_{i-4,j-2} + 4s_{i-4,j-1} + 5s_{i-4,j} + 4s_{i-4,j+1} + 3s_{i-4,j+2} + 2s_{i-4,j+3} + s_{i-4,j+4}) + \\
& (2s_{i-3,j-4} + 4s_{i-3,j-3} + 6s_{i-3,j-2} + 8s_{i-3,j-1} + 10s_{i-3,j} + 8s_{i-3,j+1} + 6s_{i-3,j+2} + 4s_{i-3,j+3} + 2s_{i-3,j+4}) + \\
& (3s_{i-2,j-4} + 6s_{i-2,j-3} + 9s_{i-2,j-2} + 12s_{i-2,j-1} + 15s_{i-2,j} + 12s_{i-2,j+1} + 9s_{i-2,j+2} + 6s_{i-2,j+3} + 3s_{i-2,j+4}) + \\
& (4s_{i-1,j-4} + 8s_{i-1,j-3} + 12s_{i-1,j-2} + 16s_{i-1,j-1} + 20s_{i-1,j} + 16s_{i-1,j+1} + 12s_{i-1,j+2} + 8s_{i-1,j+3} + 4s_{i-1,j+4}) + \\
& (5s_{i,j-4} + 10s_{i,j-3} + 15s_{i,j-2} + 20s_{i,j-1} + 25s_{i,j} + 20s_{i,j+1} + 15s_{i,j+2} + 10s_{i,j+3} + 5s_{i,j+4}) + \\
& (4s_{i+1,j-4} + 8s_{i+1,j-3} + 12s_{i+1,j-2} + 16s_{i+1,j-1} + 20s_{i+1,j} + 16s_{i+1,j+1} + 12s_{i+1,j+2} + 8s_{i+1,j+3} + 4s_{i+1,j+4}) + \\
& (3s_{i+2,j-4} + 6s_{i+2,j-3} + 9s_{i+2,j-2} + 12s_{i+2,j-1} + 15s_{i+2,j} + 12s_{i+2,j+1} + 9s_{i+2,j+2} + 6s_{i+2,j+3} + 3s_{i+2,j+4}) + \\
& (2s_{i+3,j-4} + 4s_{i+3,j-3} + 6s_{i+3,j-2} + 8s_{i+3,j-1} + 10s_{i+3,j} + 8s_{i+3,j+1} + 6s_{i+3,j+2} + 4s_{i+3,j+3} + 2s_{i+3,j+4}) + \\
& (s_{i+4,j-4} + 2s_{i+4,j-3} + 3s_{i+4,j-2} + 4s_{i+4,j-1} + 5s_{i+4,j} + 4s_{i+4,j+1} + 3s_{i+4,j+2} + 2s_{i+4,j+3} + s_{i+4,j+4})
\end{aligned}
$$

Furthermore, generating filtered signal samples outputs P^{11x1}, P^{1x11}, and P^{11x11}, produces the following relationships:

$$P_{i,j}^{11x1} = \quad s_{i-5,j} + 2s_{i-4,j} + 3s_{i-3,j} + 4s_{i-2,j} + 5s_{i-1,j} + 6s_{i,j} + 5s_{i+1,j} + 4s_{i+2,j} + 3s_{i+3,j} + 2s_{i+4,j} + s_{i+5,j}$$

$$P_{i,j}^{1x11} = \quad s_{i,j-5} + 2s_{i,j-4} + 3s_{i,j-3} + 4s_{i,j-2} + 5s_{i,j-1} + 6s_{i,j} + 5s_{i,j+1} + 4s_{i,j+2} + 3s_{i,j+3} + 2s_{i,j+4} + s_{i,j+5}$$

$$\begin{aligned}
P_{i,j}^{11x11} = (&s_{i-5,j-5} + 2s_{i-5,j-4} + 3s_{i-5,j-3} + 4s_{i-5,j-2} + 5s_{i-5,j-1} + 6s_{i-5,j} + \\
&5s_{i-5,j+1} + 4s_{i-5,j+2} + 3s_{i-5,j+3} + 2s_{i-5,j+4} + s_{i-5,j+5}) + \\
(&2s_{i-4,j-5} + 4s_{i-4,j-4} + 6s_{i-4,j-3} + 8s_{i-4,j-2} + 10s_{i-4,j-1} + 12s_{i-4,j} + \\
&10s_{i-4,j+1} + 8s_{i-4,j+2} + 6s_{i-4,j+3} + 4s_{i-4,j+4} + 2s_{i-4,j+5}) + \\
(&3s_{i-3,j-5} + 6s_{i-3,j-4} + 9s_{i-3,j-3} + 12s_{i-3,j-2} + 15s_{i-3,j-1} + 18s_{i-3,j} + \\
&15s_{i-3,j+1} + 12s_{i-3,j+2} + 9s_{i-3,j+3} + 6s_{i-3,j+4} + s_{i-3,j+5}) + \\
(&4s_{i-2,j-5} + 8s_{i-2,j-4} + 12s_{i-2,j-3} + 16s_{i-2,j-2} + 20s_{i-2,j-1} + 24s_{i-2,j} + \\
&20s_{i-2,j+1} + 16s_{i-2,j+2} + 12s_{i-2,j+3} + 8s_{i-2,j+4} + 4s_{i-2,j+5}) + \\
(&5s_{i-1,j-5} + 10s_{i-1,j-4} + 15s_{i-1,j-3} + 20s_{i-1,j-2} + 25s_{i-1,j-1} + 30s_{i-1,j} + \\
&25s_{i-1,j+1} + 20s_{i-1,j+2} + 15s_{i-1,j+3} + 10s_{i-1,j+4} + 5s_{i-1,j+5}) + \\
(&6s_{i,j-5} + 12s_{i,j-4} + 18s_{i,j-3} + 24s_{i,j-2} + 30s_{i,j-1} + 36s_{i,j} + 30s_{i,j+1} + \\
&24s_{i,j+2} + 18s_{i,j+3} + 12s_{i,j+4} + 6s_{i,j+5}) + \\
(&5s_{i+1,j-5} + 10s_{i+1,j-4} + 15s_{i+1,j-3} + 20s_{i+1,j-2} + 25s_{i+1,j-1} + 30s_{i+1,j} + \\
&25s_{i+1,j+1} + 20s_{i+1,j+2} + 15s_{i+1,j+3} + 10s_{i+1,j+4} + 5s_{i+1,j+5}) + \\
(&4s_{i+2,j-5} + 8s_{i+2,j-4} + 12s_{i+2,j-3} + 16s_{i+2,j-2} + 20s_{i+2,j-1} + 24s_{i+2,j} + \\
&20s_{i+2,j+1} + 16s_{i+2,j+2} + 12s_{i+2,j+3} + 8s_{i+2,j+4} + 4s_{i+2,j+5}) + \\
(&3s_{i+3,j-5} + 6s_{i+3,j-4} + 9s_{i+3,j-3} + 12s_{i+3,j-2} + 15s_{i+3,j-1} + 18s_{i+3,j} + \\
&15s_{i+3,j+1} + 12s_{i+3,j+2} + 9s_{i+3,j+3} + 6s_{i+3,j+4} + 3s_{i+3,j+5}) + \\
(&2s_{i+4,j-5} + 4s_{i+4,j-4} + 6s_{i+4,j-3} + 8s_{i+4,j-2} + 10s_{i+4,j-1} + 12s_{i+4,j} + \\
&10s_{i+4,j+1} + 8s_{i+4,j+2} + 6s_{i+4,j+3} + 4s_{i+4,j+4} + 2s_{i+4,j+5}) + \\
(&s_{i+5,j-5} + 2s_{i+5,j-4} + 3s_{i+5,j-3} + 4s_{i+5,j-2} + 5s_{i+5,j-1} + 6s_{i+5,j} + \\
&5s_{i+5,j+1} + 4s_{i+5,j+2} + 3s_{i+5,j+3} + 2s_{i+5,j+4} + s_{i+5,j+5})
\end{aligned}$$

Mathematical manipulation may be employed to produce the following:

$$\begin{aligned}
P_{i,j}^{11x11} = &(P_{i-1,j-1}^{9x9} + P_{i-1,j+1}^{9x9} + P_{i+1,j-1}^{9x9} + P_{i+1,j+1}^{9x9}) \\
&-(P_{i,j-3}^{11x1} + 2P_{i,j-2}^{11x1} + 3P_{i,j-1}^{11x1} + 2P_{i,j}^{11x1} + 3P_{i,j+1}^{11x1} + 2P_{i,j+2}^{11x1} + P_{i,j+3}^{11x1}) \\
&-(P_{i-3,j}^{1x11} + 2P_{i-2,j}^{1x11} + 3P_{i-1,j}^{1x11} + 2P_{i,j}^{1x11} + 3P_{i+1,j}^{1x11} + 2P_{i+2,j}^{1x11} + P_{i+3,j}^{1x11}) \\
&-P_{i,j}^{7x7} + 2(P_{i,j}^{7x1} + P_{i,j}^{1x7}) - 4s_{i,j}
\end{aligned} \qquad [1]$$

Equation [1] above illustrates that a direct two-dimensional pyramid filter architecture of order 2N-1, in this case where N is six, may potentially be implemented using either four two-dimensional pyramid filters of order [2(N-1)-1], that is nine, or one two-dimensional pyramid filter of order [2(N-1)-1] using four signal sample matrices $P^{9x9}_{i-1,j-1}, P^{9x9}_{i-1,j+1}, P^{9x9}_{i+1,j-1}, P^{9x9}_{i+1,j+1}$ and fourteen one-dimensional pyramid filters of order 2N-1, here eleven, the filters being row-wise and column-wise, in this example. It also employs one two-dimensional pyramid filter of order [2(N-2)-1], that is seven here, to produce $P^{7x7}_{i,j}$ and two one-dimensional pyramid filters of order [2(N-2)-1], that is seven here, to produce two signal sample matrices $P^{7x1}_{i,j}, P^{1x7}_{i,j}$, in this example. FIG. 3 is a schematic diagram illustrating such an embodiment, although, of course, the claimed subject matter is not limited in scope to this particular implementation or embodiment. For example, the output signal samples corresponding to those produced by four two-dimensional pyramid filters of order 2(N-1)-1, here order nine where N is six, and also the output signal samples produced by two-dimensional pyramid filter of order 2(N-2)-1, here order seven, may not necessarily be produced by two-dimensional pyramid filters. As just one example, these output signals may be produced using one-dimensional pyramid filters. One such filter is shown in FIG. 2, although, again, additional approaches to producing the output signals for the architecture shown in FIG. 3 may also be employed.

FIG. 3 illustrates an integrated circuit (IC), 300, although, of course, alternative embodiments may not necessarily be implemented on a single integrated circuit chip. IC 300 includes a two-dimensional pyramid filter architecture of an order 2N-1, where N is a positive integer greater than five, here six, in operation, capable of producing, on respective clock cycles, at least the following. Pyramid filtered output signals are produced corresponding to output signals produced by fourteen one-dimensional pyramid filters of order 2N-1, again, eleven in this example where N is six, 330, 332, 334, 340, 342, 344, 350, 352, 354, 360, 362, 364, 366, and 368 in FIG. 3. Pyramid filtered output signals are also produced corresponding to output signals produced either by four two-dimensional pyramid filters or one two-dimensional pyramid of order [2(N-1) – 1] or nine here, where N is six, using signal sample matrices $P_{i-1,j-1}^{9x9}, P_{i-1,j+1}^{9x9}, P_{i+1,j-1}^{9x9}, P_{i+1,j+1}^{9x9}$. These output signals are summed by adder 310 in FIG. 3. Pyramid filtered output signals are also produced corresponding to output signals produced by one two-dimensional pyramid filter of order [2(N-2) – 1] or seven here, where N is six , using signal sample matrix $P_{i,j}^{7x7}$ and two one-dimensional pyramid filter of order 2(N-2)-1 using signal sample matrices $P_{i,j}^{7x1}, P_{i,j}^{1x7}$. These three output signals $P_{i,j}^{7x7}$, $P_{i,j}^{7x1}, P_{i,j}^{1x7}$ and the input signal $s_{i,j}$ are summed by adder 390 in FIG. 3. Likewise, the respective output signals in this two dimensional pyramid filter architecture implementation, in the implementation in FIG. 3, for example, the output signals of 330, 332, 334, 340, 342, 344, 350, 352, 354, 360, 362, 364, 366, and 368 are summed on respective clock cycles of the two dimensional pyramid filter architecture, by adders 370 and 375 in FIG. 3. Adder 380 sums the output signals of 310, 370, 375 and 390. Of course, FIG. 3 is just one possible example of an implementation and the claimed subject matter is not limited in scope to this or to another particular implementation.

For example, N is not limited to six. Likewise, the pyramid filtered output signals that correspond to output signals produced by a two-dimensional pyramid filter are not limited to being implemented by one-dimensional pyramid filters or to two-dimensional pyramid filters. Likewise, as previously indicated, if one-dimensional filters are employed, then the filters are not limited to the implementation approach described in aforementioned U. S. Patent Application Serial No. 09/754,684, titled “Multiplierless Pyramid Filter,” filed January 3, 2001, by Tinku Acharya, or in aforementioned U. S. Patent Application Serial No., titled "Pyramid Filter," (attorney docket 042390.P11211), filed on March 28, 2001, by Tinku Acharya. For example, one-dimensional pyramid filters other than multiplierless pyramid filters may be employed. Likewise, depending on the implementation, different numbers of such pyramid filters and different orders of such pyramid filters may be employed. For example, the output signals may be combined or processed in a way to produce pyramid filtered output signals corresponding to pyramid filters of a different number, dimension, or order.

It will, of course, be understood that, although particular embodiments have just been described, the claimed subject matter is not limited in scope to a particular embodiment or implementation. For example, one embodiment may be in hardware, whereas another embodiment may be in software. Likewise, an embodiment may be in firmware, or any combination of hardware, software, or firmware, for example. Likewise, although the claimed subject matter is not limited in scope in this respect, one embodiment may comprise an article, such as a storage medium. Such a storage medium, such as, for example, a CD-ROM, or a disk, may have stored thereon instructions, which when executed by a system, such as a computer system or platform, or an imaging system, for example, may result in an embodiment of a method in accordance with the claimed subject matter being executed, such as an embodiment of a method of filtering or processing an image or video, for example, as previously described. For example, an image processing platform or an imaging processing system may include an image processing unit, a video or image input/output device and/or memory.

While certain features of the claimed subject matter have been illustrated and described herein, many modifications, substitutions, changes and equivalents will now occur to those skilled in the art. It is, therefore, to be understood that the appended claims are intended to cover all such modifications and changes as fall within the true spirit of the claimed subject matter.